UNIVERSITY OF EDINBURGH
Edinburgh Regional Computing Centre

User Note 66

June 1987

Title:

A SCRIBE PRIMER

Author: R.J.Hare

Contact: Your User Support Team

Software Support Category: See Note 15

Synopsis

This document introduces the use of version 3.7 of the local implementation of the SCRIBE document preparation program. This implementation is available on all mainframe machines controlled by the Computing Service and running the EMAS-3 operating system, and is designed to aid the preparation and production of various types of document.

The document is not intended to be an exhaustive SCRIBE manual, the locally produced SCRIBE reference manual, [7] and the manual provided by Unilogic [11] are **essential reading** for any users intending to use SCRIBE for anything other than the simplest of jobs. Reference copies of the latter document are available for inspection in Computing Service offices at 59 George Square and the James Clerk Maxwell Building, and copies may be purchased at cost price from the same locations.

The document also contains a short introduction to some of the terms used in typography, and some hints on how to achieve an aesthetically pleasing document layout.

Prepared using SCRIBE and printed on a Xerox X2700 laser printer

Keywords

Document formatting, SCRIBE, Text formatting

Edinburgh Regional Computing Centre

James Clerk Maxwell Building, The King's Buildings, Mayfield Road, Edinburgh, EH9 3JZ. Telephone 031-667 1081

Table of Contents

1 Introduction

Background

SCRIBE is a sophisticated program for the preparation of documents in several forms.

It is a local implementation of the well known SCRIBE Document Preparation System, marketed by Unilogic Inc. of Pittsburgh.

Unilogic SCRIBE is implemented locally on the Computing Service VAX/VMS machine. Access to this implementation of Unilogic SCRIBE is available to suitably demanding users, though it is envisaged that the local implementations described here and elsewhere should adequately satisfy the requirements of most users most of the time.

SCRIBE was written by Brian K. Reid, and was originally distributed to DECsystem-10 and DECsystem-20 users free through the auspices of the DEC User's Society.

Eventually, the program was acquired by Unilogic who now handle the marketing and maintenance of SCRIBE on an extended range of machines and operating systems.

The SCRIBE instruction format has been accepted by the Computing Service as a suitable standard for the preparation of input (manuscript) files for subsequent formatting by SCRIBE into finished output (document) files.

Unfortunately, no implementation of SCRIBE is available from Unilogic which will run on Amdahl machines. Accordingly, a local implementation of SCRIBE has been written. This implementation does not provide full Unilogic SCRIBE functionality, but, as implied above, it is intended that it satisfy the requirements of most users for most of the time.

Design Philosophy

Unlike many other mainframe text processing programs, SCRIBE's design philosophy is oriented to the idea that the input (manuscript) file to be processed to produce a finished output (document) file should **not** contain formatting instructions whose effect is immutable, but which instead specify the required output format in some general way.

SCRIBE itself will then interpret these generalised instructions, and will generate an output (document) file formatted for the specified document type and output device type.

The necessarily detailed information controlling the interpretation of individual instructions for each document type and output device type is held in a SCRIBE database and the casual user need never know of the existence of this database.

For a fuller exposition of SCRIBE's underlying design philosophy, the user should consult the SCRIBE User Manual [11] and the listed review article [5] which describes SCRIBE in the context of many other mainframe text processors currently in use.

This design philosophy has been largely incorporated in the locally written implementation of SCRIBE described here.

The article by Browning [4] describes briefly some of the problems associated with computerised document preparation and production, though not in the context of SCRIBE.

Overview of Functionality

SCRIBE allows the user to produce on the computer documents of various types formatted for a variety of output device types, incorporating within the documents a variety of text 'objects' or 'environments' and typefaces.

Several standard document types are built into the SCRIBE system, and sophisticated users may set up their own document types, either from scratch or, by modifying existing types as required. The document types supplied as standard are TEXT (the default), ARTICLE, REPORT, MANUAL and LETTER.

Within each document type there exist standard 'environments' which are formatted automatically by SCRIBE according to both the type of document being processed and the output device type requested. The demanding user may set up new environments, or modify existing ones as required. The standard environments include, VERSE, ENUMERATE, QUOTATION, etc.

SCRIBE includes fully automatic sub-systems for setting up pagination and, for appropriate document types, section numbering, indexes and tables of contents.

SCRIBE is capable of formatting input (manuscript) files, and producing output (document) files for four output device types at present, namely the line printer (LPT), the Phillips/Mullard GP300 dot-matrix printer (GP300), a terminal file (FILE) and the Xerox X2700 laser printer (X2700). There are also two 'dummy' drafting device types available (DRAFTX2700 and DRAFTGP300) which produce line-printer files which have the same page and line structure as files produced for device types X2700 and GP300.

SCRIBE also has sophisticated error handling facilities, and diagnostic error messages are displayed online as the errors occur. These messages are also stored in a 'housekeeping' file created by SCRIBE which may be inspected at leisure when the job is complete. This file will be named according to the first six characters of the input (manuscript) file, and will have the 'extension' £ERR.

2 Running SCRIBE

SCRIBE forms part of the standard EMAS subsystem, and is therefore available to all users automatically. SCRIBE does however require large amounts of internal storage space when processing large jobs, and in order to accomodate this requirement, your maximum temporary file size should be increased. This should be done with the command:

```
Command:OPTION USTACKSIZE=256
```

This procedure is a once and for all operation and does not need to be re-entered every time you log on or run SCRIBE. For full details of the OPTION command, please consult the EMAS-3 User's Guide. [6]

You may now run SCRIBE by entering the command:

Command:SCRIBE **manuscript,document**

where **manuscript** is the name of the input (manuscript) file which contains the text to be processed and the embedded SCRIBE instructions, and **document** is the processed output (document) file, formatted for the required device type.

SCRIBE's consumption of resources is heavy, so it may well be that it will be necessary to DETACH any given job to run in the background stream. Full details of the use of the DETACH command are given in the EMAS-3 User's Guide. [6]

The criteria to be considered when deciding whether to run a SCRIBE job 'live' or in the background stream are:

- the length of the document
- the output device type being used

Briefly, the longer the document, and the more sophisticated the output device type, the longer will be the amount of CPU time consumed by the job.

The time consumed by a given job may be assessed by using the METER command (please consult the EMAS-3 User's Guide [6] for details) immediately before and after the job concerned, and any job consuming more than about 30 seconds of CPU time should certainly be run in the background mode.

3 Using SCRIBE

Because of its flexibility, use of SCRIBE may be just as simple or as complicated as the user requires within the wide range of SCRIBE's functionality.

In its simplest form, use of SCRIBE will consist of merely typing in the manuscript for a simple document, with blank lines between paragraphs and perhaps section titles in upper case letters - that is with no SCRIBE instructions in the file whatsoever. If such a file is processed by SCRIBE, the result will be a paginated, filled and justified output (document) file, with line widths, top and bottom margins etc. selected according to the database entries for the line printer (LPT) which is SCRIBE's default output device type. The document type will be the default TEXT.

At the other extreme, the sophisticated user may for example be producing a large document such as a book with many chapters. In this case, the input (manuscript) file will consist of the text to be processed intermingled with whatever SCRIBE instructions are necessary to produce the desired effect.

SCRIBE instructions are preceded by an '@' character and may be incorporated intimately in the text, or placed on separate lines as required. In practice, it is convenient to embed instructions which only have a local effect intimately within the text, and those which have an effect extending over more than one line on a line of their own (the example illustrated in section 4 below should make this point clearer).

Whichever of these approaches is adopted, SCRIBEs sophisticated house-keeping will ensure that if a new chapter, or a chapter with a changed number of pages (for example) is inserted, it will not be necessary for the pagination of subsequent chapters to be updated 'by hand'. This will be done automatically by SCRIBE.

Whichever of these approaches is adopted, the following diagrammatic illustration of the process of creating a SCRIBE input (manuscript) file, processing it through SCRIBE to produce an output (document) file, and finally printing the finished document on a device of the selected type should make the whole process clearer.

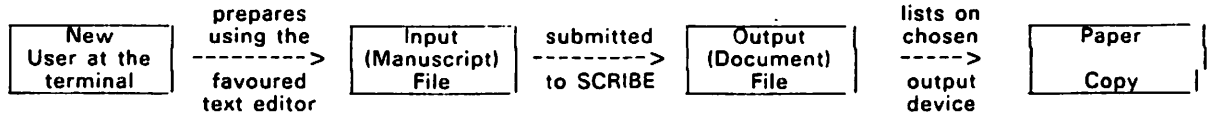

Figure 3.1 Creating a document with SCRIBE

Basically the procedure is an iterative one:

1. prepare/correct the manuscript file using the editor of your choice
2. process the manuscript through SCRIBE to produce a formatted document
3. list the document on a printer of the necessary type
4. if the document is satsfactory, go on your way rejoicing; otherwise return to step 1. and make any necessary corrections before repeating steps 2. to 4.

If you are using a microcomputer to prepare your manuscript file 'off-line', you will of course need to transfer the file from the microcomputer to an EMAS host before proceeding to step 2. In order to do this, you will need to familiarise yourself with one of the relevant file transfer programs (KERMIT or XTALK). Discussion of these programs is outwith the scope of this document, but they are fully described in references [1], [2], [3] and [8].

If you **are** using a microcomputer to prepare the manuscript file you should:

BEWARE!

Many editor/word-processors on microcomputers insert non-standard sequences of non-printing characters into files created using these programs (typically, 'soft' returns or sequences used to represent non-standard characters such as a £). If presented to SCRIBE, such non-standard sequences of non-printing characters (usually characters whose ASCII code is greater than 127) will be ignored by SCRIBE and a suitable warning message will be displayed. You should ensure that you operate your editor/word processor in 'plain text' mode to avoid such problems, and even in this circumstance, ensure that you enter into your manuscript file the recognised SCRIBE instruction for obtaining any 'non-standard' characters, **not** the sequence which may be required by the word-processor in use (for example, the £ character in SCRIBE is obtained by typing `@T[` **shifted 3** `]`, **whatever** the character in the 'upper case 3' position may be on the keyboard being used). In particular, you should **never** 'format' the manuscript on the microcomputer before transferring the file to the mainframe (what's the point? you are going to format it with SCRIBE anyway!).

If in doubt - seek advice!

4 Example

Perhaps the best way to introduce the new user to SCRIBE is to consider its use in a simple example, explaining briefly the use of the instructions as they are encountered within the input (manuscript) file.

The example, shown on the following two pages consists of: on the first page, a simple SCRIBE input (manuscript) file, and on the second page the final copy produced by printing the resultant output (document) file.

Both the input and the output should be studied carefully in conjunction with one another and with the aid of the immediately following explanatory notes.

This should give some insight into how a SCRIBE input (manuscript) file is structured.

Note that although in the following example, SCRIBE instructions are capitalised for purposes of emphasis, SCRIBE instructions may be entered in either upper or lower case.

```
@MAKE[TEXT]                                          ;see note 1
@DEVICE[X2700]                                       ;see note 2
@MAJORHEADING[Memo]                                  ;see note 3
@CENTRE[To all EMAS, BUSH  and EMAS-A Users]         ;see note 4
@HEADING[SCRIBE Service on EMAS, BUSH and EMAS-A]    ;see note 5
This memo serves to announce the introduction of a SCRIBE service on
EMAS, BUSH and EMAS-A.
SCRIBE as implemented on EMAS, BUSH and EMAS-A has a significant
proportion of the functionality of Unilogic SCRIBE (which is publicly
available on the Computing Service VAX/VMS machine).

For the benefit of the uninitiated, SCRIBE is a sophisticated text
processing program which allows the user to extend considerable control
over many of the aspects of a document being prepared on the computer.

These include:
@BEGIN[ITEMIZE]                                      ;see note 6
choice of output device type, eg: LPT, GP300, X2700, etc.

choice of document type, eg: MANUAL, REPORT, LETTER, etc.

choice of different text 'environments' within document types,
eg: TEXT, EXAMPLE, DISPLAY, etc.

choice of typeface, eg: @B[bold], @G[greek],         ;see note 7
@C[small capitals], etc.

automatic pagination, section numbering and where appropriate
contents page generation

automatic indexing
@END[ITEMIZE]                                        ;see note 6
It is hoped that SCRIBE will become the text processor of choice on
EMAS, BUSH and EMAS-A for the forseeable future, and, with this hope in
mind, local documentation is in the process of being prepared.
This will comprise:
@BEGIN[ENUMERATE]                                    ;see note 8
A 'primer' serving as a brief introduction to SCRIBE, its design
philosophy and use

A 'reference manual' giving a full description of all the facilities
available in SCRIBE as implemented on EMAS, BUSH and EMAS-A.
@END[ENUMERATE]                                      ;see note 8
Both of these documents will however need to be read in conjunction
with the SCRIBE User Manual [11] by any user intending to
become a SCRIBE 'wizard', particularly those users who will need
to modify the default settings for line spacing, facecode and
general document layout etc.
@FLUSHRIGHT[Roger Hare                               ;see note 9
Computing Service
@VALUE[DATE]]                                        ;see notes 9, 10 and 11
```

Memo

To all EMAS, BUSH and EMAS-A Users

SCRIBE Service on EMAS, BUSH and EMAS-A

This memo serves to announce the introduction of a SCRIBE service on EMAS, BUSH and EMAS-A. SCRIBE as implemented on EMAS, BUSH and EMAS-A has a significant proportion of the functionality of Unilogic SCRIBE (which is publicly available on the Computing Service VAX/VMS machine).

For the benefit of the uninitiated, SCRIBE is a sophisticated text processing program which allows the user to extend considerable control over many of the aspects of a document being prepared on the computer.

These include:

- choice of output device type, eg: LPT, GP300, X2700, etc.
- choice of document type, eg: MANUAL, REPORT, LETTER, etc.
- choice of different text 'environments' within document types, eg: TEXT, EXAMPLE, DISPLAY, etc.
- choice of typeface, eg: **bold**, GREEK, SMALL CAPITALS, etc.
- automatic pagination, section numbering and where appropriate contents page generation
- automatic indexing

It is hoped that SCRIBE will become the text processor of choice on EMAS, BUSH and EMAS-A for the forseeable future, and, with this hope in mind, local documentation is in the process of being prepared. This will comprise:

1. A 'primer' serving as a brief introduction to SCRIBE, its design philosophy and use
2. A 'reference manual' giving a full description of all the facilities available in SCRIBE as implemented on EMAS, BUSH and EMAS-A.

Both of these documents will however need to be read in conjunction with the SCRIBE User Manual [11] by any user intending to become a SCRIBE 'wizard', particularly those users who will need to modify the default settings for line spacing, facecode and general document layout etc.

Roger Hare
Computing Service
01/06/87

Explanatory Notes

1. This is an example of the @MAKE instruction which specifies the type of document to be prepared. The format is self-explanatory, namely:

   @MAKE[**documenttype**]

   where **documenttype** is the SCRIBE name for the type of document to be produced. Currently, the document types available are:

TEXT	TEXT is the default document type - that is, it is what is assumed to be required if no @MAKE instruction is encountered in the input (manuscript) file. TEXT will be processed to produce filled justified paragraphs, with the first line indented, and with paragraphs separated by blank lines.
ARTICLE	As TEXT, with the addition of; numbered SECTIONS, SUBSECTIONS, PARAGRAPHS and APPENDIXes; unnumbered PREFACESECTION; numbered THEOREMS, LEMMAS, etc on numbered pages. A table of contents is generated automatically.
REPORT	as ARTICLE with the addition of numbered CHAPTERS.
MANUAL	as REPORT with the addition of an index.
LETTER	Produces a letter. LETTER has several dedicated environments such as ADDRESS, GREETING, RETURNADDRESS, BODY etc.

2. This is an example of the SCRIBE @DEVICE instruction which specifies the device type for which the output (document) file is to be formatted. It has the form:

   @DEVICE[**devicetype**]

   where **devicetype** is the SCRIBE device type name. Currently, the device typess available are:

FILE	unpaged terminal file. The file produced will not incorporate any bold-facing, underlining, superscripting etc., but is ideal for quick 'proof-reading' at the terminal.
LPT	line printer. This is the default device, that is the device which is selected if no @DEVICE instruction is present in the input (manuscript) file.
GP300	The Phillips/Mullard GP300 dot-matrix printer (Warning: there is a real money charge for using this device, please contact your Support Team for details of current charges). DRAFTGP300 A line-printer compatible 'drafting' version of the above.
X2700	The Xerox X2700 laser printer (Warning: there is a real money charge for using this device, please contact your Support Team for details of current charges.
DRAFTX2700	A line-printer compatible 'drafting' version of the above.

3. This is an example of a SCRIBE @MAJORHEADING instruction which has the form:

   @MAJORHEADING[**headingtext**]

   where **headingtext** is the text to be used as a heading. Other SCRIBE titling or sectioning instructions are; @HEADING and @SUBHEADING which are unnumbered and do not appear in the table of contents; @PREFACESECTION and @UNNUMBERED which are unnumbered and do appear in the table of contents, and; @CHAPTER, @SECTION, @SUBSECTION, @PARAGRAPH, @APPENDIX and @APPENDIXSECTION which are all numbered and do appear in the table of contents (Note: The detailed

appearence of all environments may vary slightly when different device typess and/or document types are used. You may need to experiment a little to find the document type which best suits your requirements.).

4. This is an example of the use of the unfilled CENTRE environment in the form:

   @**environmentname**[text to be processed]

   where **environmentname** is the name of the SCRIBE environment. There are many environments within SCRIBE, a few of which are described briefly in the following notes.
5. This is an example of another of the SCRIBE heading instructions described briefly in note 5.
6. This is an example of another of SCRIBE's environments, this time a filled one. ITEMIZE produces an itemised list, each item being marked with a 'bullet' character (in this case a hyphen). Note the slightly different form which may be used as an alternative to that illustrated above for an excursion into any SCRIBE environment:

   @BEGIN[**environmentname**]
   .
   text to be processed
   .
   @END[**environmentname**]

   Note also that although in the processed output, the items are not separated from one another by any blank space; in the input (manuscript) file each item is separated by a blank line. This is the usual way of separating items/paragraphs etc within a SCRIBE environment in the manuscript file. The separation of items in the output (document) file will usually be set to one line, but may be varied at will by the user (as in this case).

   Other filled environments are TEXT (the default, and the only SCRIBE environment which is also a document type), DESCRIPTION, ENUMERATE (see note 8 below), and QUOTATION.
7. This line contains examples of shifts into and out of some of SCRIBE's facecode environments. Illustrated are shifts into **bold**, γρεεκ and SMALL CAPITALS respectively. Other facecode environments include @I (italic), @P (bold italic), @U (underline), etc.
8. This is an example of a different kind of list (see note 6 above). The ENUMERATEd list has each item set off from the surrounding text with a numeral rather than a 'bullet' character. ENUMERATE is another example of a SCRIBE filled environment. Both ENUMERATEd and ITEMIZEd lists may be nested to the required level, each level having a different 'bullet' character or numbering style. The 'bullet' character and numbering style repeat after the second and third levels of nesting respectively.
9. This is an example of the unfilled FLUSHRIGHT environment. The name is more or less self explanatory, each line in the input (manuscript) file is flushed against the right hand margin in the output (document) file. There is also a FLUSHLEFT environment. Other unfilled environments are FORMAT which is similar to FLUSHLEFT; VERBATIM which is similar to FLUSHLEFT but uses a fixed-width typeface; and PROGRAMEXAMPLE whose use is obvious from its name.
10. This is an example of the use of the @VALUE instruction. The argument to the @VALUE instruction may either be a user-defined string set up with an @STRING instruction:

    @STRING[**name="text"**]

where **name** is the users name for the string **text**, or a predefined string, several of which exist within SCRIBE. DATE is a predefined string, others are TIME, YEAR, TIMESTAMP, SECTIONNUMBER, SECTIONTITLE etc.

11. Note the nesting of one SCRIBE environment or instruction within another environment (the @VALUE instruction within the @FLUSHRIGHT environment). This is quite acceptable within SCRIBE, and generally, any instruction or environment may be nested within another environment. The nesting of environments in particular may be carried to an arbitrary level, but it is the responsibility of the user to ensure that such nesting is actually sensible!

5 Where To Go From Here?

Well, the first thing you should do is to try using SCRIBE on a short simple file - an internal memo say, or a handout for a lecture.

Then you could try inserting SCRIBE instructions one at a time, starting with say an @DEVICE[GP300] or @DEVICE<X2700> instruction in order that you can see how much nicer the output from a high quality printer is than that from a line printer (though in general, it is good practice to use the line printer or a terminal file for all but the final drafts of any document). In order to see a sample of the character sets (fonts available) with the GP300 printer, please consult User Note 50 [9] - at this stage you are **not** recommended to try changing fonts within SCRIBE other than by using the facecode instructions - this is not a suitable exercise for new users.

Also, be warned that a real money charge is made on a per sheet basis for using the GP300 and X2700 printers. For details of current charges, please consult your Support Team.

As you gradually increase the complexity and number of SCRIBE instructions in your file, you should be able to see how SCRIBEs powerful facilities will enable you to produce documents formatted more or less at will.

Once you have exhausted the facilities briefly described or mentioned in passing in this primer, you should study the companion SCRIBE Reference Manual [7] and if you require even further information about SCRIBE you should graduate to the SCRIBE User Manual [11] which contains fuller descriptions of SCRIBE's facilities than are included in the locally produced Reference Manual.

The additional facilities described in the Reference Manual include:

- indexing
- cross-referencing to other sections within a document
- tabulation
- page headings and footings
- modification of existing, and, definition of new environments
- control of overall document style

If you eventually feel that SCRIBE as implemented on EMAS does not have enough facilities for your application (and this should not happen often!), you should consider using the implementations of SCRIBE on the Computing Service VAX/VMS machine. Facilities exist for the 'remote' and 'live' use of Unilogic SCRIBE on this machine. For further information on these facilities, please contact your Support Team.

6 Typography and Typesetting

SCRIBE is not a full typesetting system, neither is the best of the high-quality printers available (the laser printer) capable of producing copy which would be regarded as particularly high quality by professional printers. However, SCRIBE is capable of producing well laid out documents, and the laser printer is capable of producing reasonable quality output, so it seems reasonable to include a few words on the basic principles of typography and typesetting, so that SCRIBE users at all levels will stand at least a reasonable chance of a) understanding what their publisher is talking about; and, b) producing a document which is aesthetically pleasing to look at.

These brief notes are based largely on the relevant sections of the OUCS Manual "Getting Started on the Lasercomp" [10] suitably modified to reflect the fact that SCRIBE is being described, not the Lasercomp. We are grateful to OUCS for their permission to draw on this material, but any errors or omissions are entirely the responsibility of the present author.

Terminology

Picas and Points
Picas and points are units of distance traditionally used in the printing trade. There are 6 picas to the inch, and 12 points to the pica. It follows that there are 72 points per inch.

Ascenders, Descenders and x-Height
Ascenders and descenders are respectively those parts of a character (usually a letter) which extend above the x-height or below the baseline of a line of type, eg: the 'top' portion of a 't' or an 'h' are ascenders, and the tails of a 'y' or a 'g' are descenders.

The x-height of a particular typeface is the height of those characters which have neither ascenders nor descenders, in particular, the 'x' character.

Point size
The size (height) of type is usually expressed in points. In principle, the point size is the distance from the top of the highest ascender to the bottom of the lowest descender, though in practice the convention adopted for different typefaces may be slightly different.

Baseline
The baseline is an imaginary horizontal line upon which the characters in a line of typesetting are considered to rest.

Leading
Leading is the distance from one baseline to the next (above or below), expressed in points. Effectively this is the line spacing of the text. The leading may be the same as the nominal point size of the type, but increasing the leading can make the text easier to read. The amount of extra leading required varies according to the typeface in use and the length (measure) of the line of text, usually a leading one or two points greater than the point size will suffice. Leading and point size are often expressed together using the compound term:

"point size" on "leading"

for example, "10 on 12" meaning a 10 point typeface used with a 12 point leading or line spacing.

Leading is sometimes used to describe the **difference** between the type size and the line spacing.

Measure
The measure is effectively the length of the line in the processed text. This is often expressed in picas.

Typeface and Fonts
A typeface is a set of characters designed together, eg: Kosmos 10 point. A font is a family of typefaces designed to be used together, eg: Kosmos 10 point, Kosmos 10 point italic, Kosmos 10 point bold. The device types available from within SCRIBE do not allow the use of very many typefaces or fonts because of their limited capacities.

Filling and Justification
Filling is the process whereby the line of processed text has as many words placed on it as possible, without crossing over the boundary of the right hand margin. Justification is the process whereby each line of filled text is made the same length as every other by inserting extra white space between the words on the line in a controlled way. The end result is that both the left and right margins are straight, and (usually) that the length of the line will be equal to the measure.

Introductory Notes on Typography

Choice of Typeface
The typeface chosen for any job should as far as possible be suited to the requirements of the job. Some characteristics of different typefaces are enumerated:

1. Serif or sans-serif. A sans-serif typeface such as Kosmos should not be used in the body of a long text or in long reports. It is difficult to read and may cause eye strain.

   Having said this, bear in mind that the 'serifs' in a serif typeface may 'fill-in' when being copied, depending on the method of reproduction being used. This manual is Xeroxed, and a sans-serif typeface is used.
2. The 'set' of the typeface being used should be considered. This is determined by the width of the lower case 'm'. Some typefaces are much wider than other, this fact becoming more obvious the larger the point size used. In general, typefaces with narrow sets are to be preferred for column work. (The EMAS implementation of SCRIBE does not allow the preparation of multiple-column work.)
3. Typefaces with large x-heights require more leading.
4. Some typefaces are darker than others.
5. Some typefaces are better when used in larger sizes, and some when used in smaller sizes.
6. Many jobs require italic and bold fonts as well as the Roman font. Care should be taken when mixing fonts from different typefaces. (Remember that the local high quality printers have only a restricted range of type faces and sizes available.)

Measure

1. Type set with too long a measure is difficult to read.
2. Too short a measure may produce an unacceptable amount of white space between words.
3. A measure should ideally have an average of 65-70 characters, and a measure of less than two, or more than three alphabets (<52 characters or >78 characters) is unlikely to be acceptable for the reasons stated at 1 and 2 above.

4. A reasonable guess at an acceptable measure may be made by assuming that the average width of a character in proportionally spaced text is the average character width of the 'a-w width' of a given font. This average character width should be multiplied by 65-70 to yield a reasonable measure.

Point Size

1. 9, 10 and 11 are suitable point sizes for normal text work.
2. Beginners often make headings too large.
3. Dictionaries and catalogues are specialist publications whose requirements may differ from the norm. A small point size may be acceptable because only specific entries, not large bodies of text are being read.
4. Footnotes are usually set two points smaller than that of the main text.

Leading

1. Judicious use of leading can aid legibility. It can also alter the aesthetic effect of a typeface. The more closely a typeface is set, the blacker it appears.
2. Proportionally, more leading is required when using larger point sizes. Similarly, the wider the measure, the more leading is required. Readability is the crucial consideration, **not** getting as much text on the page as possible!
3. Large bodied typefaces usually benefit from a leading of 1.5-2 points greater than the point size of the typeface.

   As a very rough guide, the following point size/leading combinations are likely to be acceptable, 8 on 8, 8 on 9, 10 on 12, 12 on 14, 14 on 16, 14 on 17, 16 on 20, but remember that much depends on the typeface and measure in use.

Margins

Margin size is determined by the proportions of a page and also by economic considerations. There should always be a larger margin at the bottom of a page than at the top; otherwise the text appears to be 'falling off the page'. A double page is considered as an entity rather than as two single pages, so the inner margins should be slightly narrower than the outer margins. Enough space must be left on the inner margins to allow for bindings.

Spacing

Spacing in typeset work is very different from that in typescript. Generally, much less white space is needed in typeset text than in typescript, as the former is denser and any white space shows up more.

1. The first paragraph of a chapter is not usually indented.
2. The conventional way to start a new paragraph is to indent it by one em-space. This amount, less than in typescript, is considered sufficient in typeset material.

   Another way to begin a paragraph is to insert extra vertical space after the last paragraph (SCRIBE does this by default, and an extra line is the usual amount). These methods are alternatives; if the second method is used, the new paragraph should not be indented.
3. There should only be a single space after punctuation, in contrast with the two spaces usually used in typescript. Extra spacing leads to 'rivers'. Close word spacing is generally more pleasing than open word spacing.
4. Care must be taken when designing headings and subheadings, that the total space involved (that is, above, below and including the heading itself) adds up to an integral multiple of the leading of the main text. This ensures that, overall, the text will be of uniform leading. This is important in the final printing of the text; the lines of the left and right hand pages will be aligned, as will the lines

on the reverse sides of the pages.

Posters, Titling etc.
It is not advisable to try to set out a poster with all the bits of text exactly in position using SCRIBE. A better method is to set all the bits of text, then cut them out and 'paste up' the layout. This is best done by placing them on top of one sheet of paper but beneath some tracing paper. The tracing paper hides the edges of the cut-out pieces which can detract from the assessment of the layout.

Typographical Tips

1. All-capitals should not be overused. Headings and titles are easier to read in mixed case.
2. Italics should never be all capitals.
3. Typefaces should not generally be mixed in great profusion.
4. Underlining should not normally be used in typesetting. Emphasis should be obtained by the use of italic or possibly bold.
5. Letter spacing (hairline spaces inserted between the letters) improves the appearance of capitals and small capitals.
6. A single line starting a new paragraph at the bottom of a page is a 'widow'; a single line ending a paragraph at the top of a new page is an orphan. While widows are now usually accepted. orphans are considered unsightly and should be avoided whenever possible.
7. The last word on a page should never be hyphenated.

Users who wish to 'go to town', and produce documents which are as nearly as possible in a form which would be acceptable to composing room staff as 'camera ready copy' should consult a book such as that by Williamson. [12]

7 References And Further Reading

[1] Adie C.J., *User Guide for BBC Micro XTALK Version 2.2*, ERCC User Note 74, June 1987.

[2] Albert-Recht A., *An Introduction to KERMIT*, ERCC User Note 78, August 1985.

[3] Albert-Recht A. and Adie C.J., *KERMIT on EMAS-3*, ERCC user Note 94, June 1987.

[4] Browning G.K.S., *Automatic Typesetting*, University Computing, Vol **6**, No 2, pp 95-98, 1984.

[5] Furuta R., Scofield J., Shaw A., *Document Formatting Systems: Survey, Concepts and Issues, ACM Computing Surveys*, Vol **14**, No 3, pp 417-472, September 1982.

[6] Hamilton-Smith N. (editor), *EMAS-3 User's Guide*, ERCC, April 1987.

[7] Hare R.J., McArthur C.D., McKendrick A., *SCRIBE Reference Manual*, ERCC User Note 67, November 1986.

[8] Howitt A., *Portable XTALK User Guide*, ERCC User Note 91, October 1985.

[9] Murison J.M., *Phillips GP300 Printer: Character Sets*, ERCC User Note 50, June 1985.

[10] Oxford University Computing Service, *Getting Started on the Lasercomp*, June, 1983.

[11] Reid Brian K., *SCRIBE Document Preparation System - User Manual*, Unilogic Ltd., April 1984.

[12] Williamson Hugh, *Book Design - the Practice of an Industrial Craft*, University of Chicago, 1983 (3rd edition).